UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

DEPARTAMENTUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

**PROIECT BAZE DE DATE**

PROFESOR COORDONATOR:

VASILE SILVIU-LAURENȚIU

STUDENT:

STOINOIU ALEXANDRU-CRISTIAN

UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

DEPARTAMENTUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

**SISTEM DE GESTIUNE A UNEI BIBLIOTECI**

PROFESOR COORDONATOR:

VASILE SILVIU-LAURENȚIU

STUDENT:

STOINOIU ALEXANDRU-CRISTIAN

CUPRINS

1. Introducere
2. Diagrama entitate-relație
   1. Reprezentare
   2. Descrierea entităților, cheilor, relațiilor și a cardinalităților
3. Diagrama Conceptuala
   1. Reprezentare
   2. Descrierea constrângerilor de integritate
   3. Scheme rationale
4. Implementarea modelului într-un sistem de gestiune a bazelor de date
   1. Crearea tabelelor si introducerea datelor
   2. Ștergerea tabelelor si constrângerilor
5. Introducere

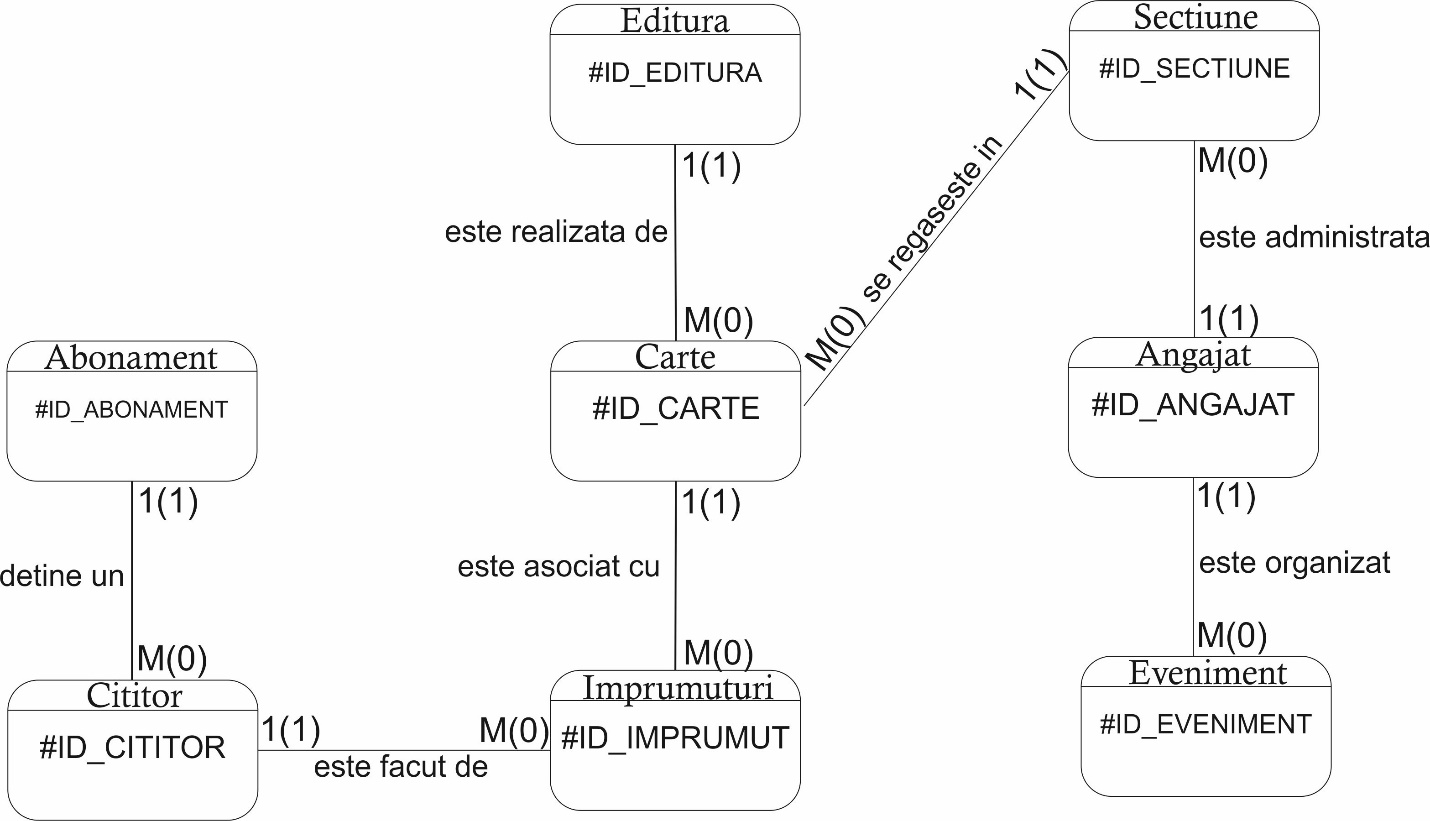
Sistemele de gestiune a bazelor de date reprezintă un instrument crucial în domeniul stocării și administrării eficiente a informațiilor. În cadrul acestui proiect, am dezvoltat o bază de date axată pe gestionarea informațiilor referitoare la mai multe biblioteci ce fac parte dintr-un sistem integrat.

Entitatea centrală a modelului de date este "Carte", care are rolul de a înregistra și gestiona informații despre cărți disponibile într-o bibliotecă. Aceasta conține detalii precum titlu, autor, editură și sectiune. Conform modelului, o carte poate fi asociată cu o editură specifică și poate fi plasată într-o anumită secțiune a bibliotecii.

Principiile fundamentale ale acestui model sunt construite pe relațiile dintre entități și pot fi enunțate astfel:

* **Un abonament poate fi detinut de un cititor sau de niciunul.**
* **Un cititor are obligatoriu un abonament.**
* **Un imprumut este facut de un cititor si doar unul.**
* **Un cititor poate face mai multe imprumuturi sau niciunul.**
* **Un imprumut este asociat unei singure carti .**
* **O carte poate fi imprumutata de mai multe ori sau niciodata.**
* **Orice carte este realizata de o editura.**
* **O editura poate redacta mai multe carti sau niciuna.**
* **Orice carte se regaseste intr-o sectiune.**
* **O sectiune poate avea mai multe carti sau niciuna.**
* **O sectiune este administrata de un singur angajat.**
* **Un angajat poate administra mai multe sectiuni sau niciuna.**
* **Un angajat poate organiza mai multe evenimente sau niciunul.**
* **Un eveniment este organizat de un singur angajat.**

1. Diagrama entitate-relatie
   1. Reprezentare



* 1. Descrierea entităților, cheilor, relațiilor și a cardinalităților

CARTE

Tabelul "Carte" reprezintă nucleul sistemului și se referă la fiecare carte înregistrată în cadrul bibliotecii. Acesta este compus dintr-un identificator unic, "ID\_Carte" (NUMBER(7)), care servește ca cheie primară a tabelului. De asemenea, tabelul dispune de chei externe care conectează entitatea la alte componente ale sistemului, precum "ID\_Editura" (NUMBER(7)) și "ID\_Sectiune" (NUMBER(7)), acestea provenind din entitățile "Editura" și "Sectiuni".

Informațiile despre fiecare carte sunt stocate în atributele "Titlu" (VARCHAR2(255)), "Autor" (VARCHAR2(50)), ce prezintă numele autorului, numele cărții, și "ID\_Editura" și "ID\_Sectiune", care indică respectiv editura la care aparține cartea și sectiunea în care este amplasată.

Această entitate reflectă relații complexe între cărți, edituri și sectiuni, furnizând o imagine detaliată și organizată a resurselor disponibile în bibliotecă.

ABONAMENTE

Tabelul "Abonamente" reprezintă entitatea ce gestionează informații legate de diverse tipuri de abonamente disponibile în cadrul bibliotecii. Cheia primară, "ID\_Abonament" (NUMBER(7)), asigură unicitatea fiecărui abonament, iar atributele "Tip\_Abonament" (VARCHAR2(50)) și "Cost\_Abonament" (DECIMAL(10, 2)) rețin detaliile referitoare la tipul și costul abonamentului.

CITITORI

Tabelul "Cititori" este dedicat înregistrării datelor despre cititorii bibliotecii. Fiecare cititor este identificat de "ID\_Cititor" (NUMBER(7)), iar atribute precum "Nume\_Cititor" (VARCHAR2(100)), "Adresa" (VARCHAR2(255)), "Email" (VARCHAR2(100)), și "ID\_Abonament" (NUMBER(7)) furnizează informații esențiale despre identitatea și abonamentul fiecărui cititor. Cheia externă "ID\_Abonament" este legată de cheia primară a tabelului "Abonamente".

IMPRUMUTURI

Tabelul "Imprumuturi" stochează informații legate de cărțile împrumutate de cititori. Cheia primară "ID\_Imprumut" (NUMBER(7)) identifică fiecare împrumut, iar atribute precum "ID\_Cititor" (NUMBER(7)), "ID\_Carte" (NUMBER(7)), "Data\_Imprumut" (DATE), și "Data\_Restituire" (DATE) rețin detalii despre cititorul implicat, cartea împrumutată și perioada de împrumut.

EDITURA

Tabelul "Editura" gestionează informații referitoare la editurile cărților din bibliotecă. Cheia primară, "ID\_Editura" (NUMBER(7)), identifică fiecare editură în mod unic, iar atributele "Nume\_Editura" (VARCHAR2(100)) și "Tara\_Sediu" (VARCHAR2(50)) conțin detalii despre numele și țara de origine a editurii.

SECTIUNI

Tabelul "Sectiuni" reflectă detaliile referitoare la diferitele sectiuni ale bibliotecii. Cheia primară, "ID\_Sectiune" (NUMBER(7)), identifică în mod unic fiecare sectiune, iar "Nume\_Sectiune" (VARCHAR2(100)) și "ID\_AngajatResponsabil" (NUMBER(7)) conțin informații despre numele și angajatul responsabil al sectiunii. Cheia externă "ID\_AngajatResponsabil" este conectată la cheia primară a tabelului "Angajati\_B".

ANGAJATI\_B

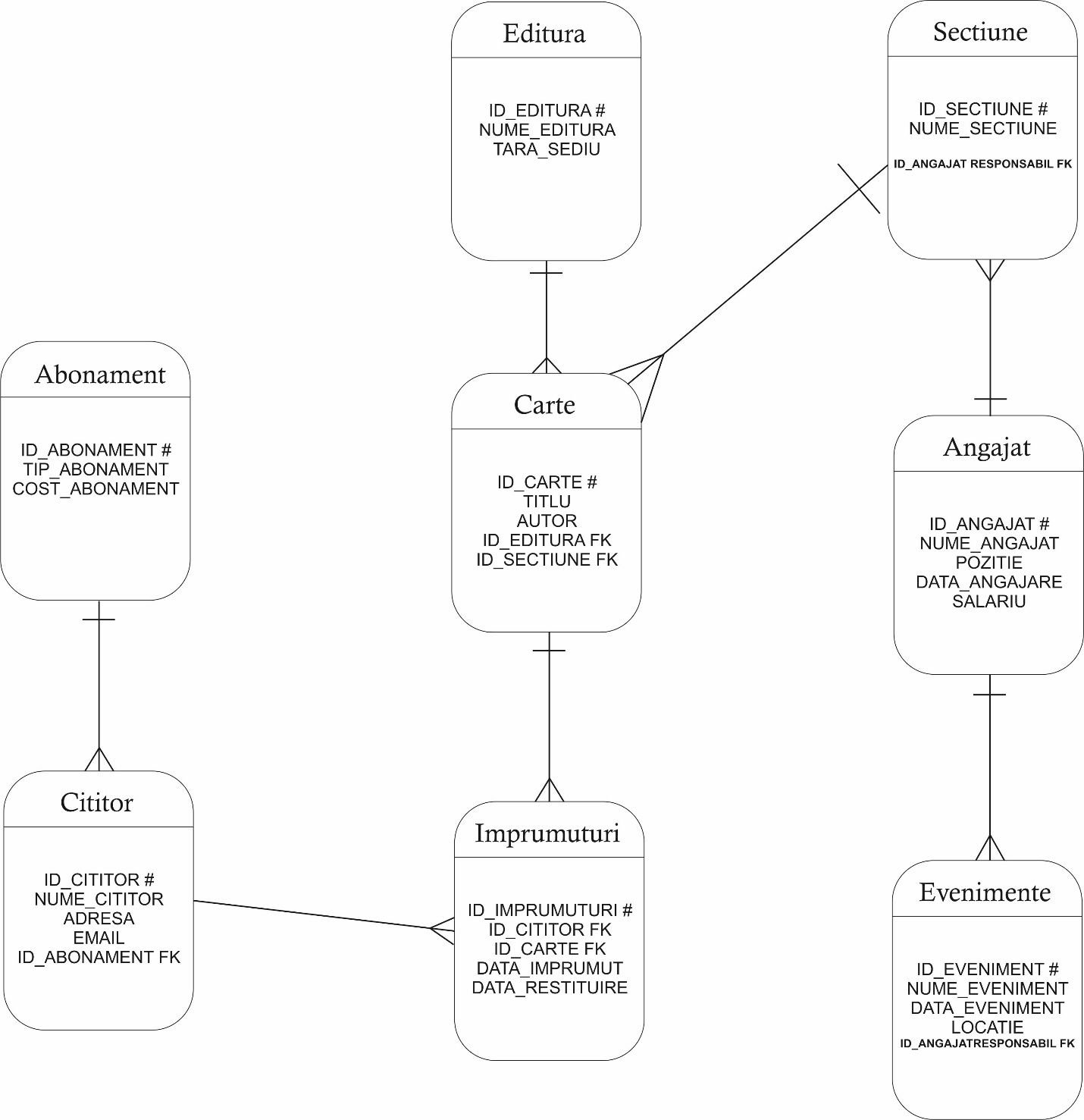
Tabelul "Angajati\_B" reprezintă entitatea ce gestionează detaliile despre angajații bibliotecii. Fiecare angajat este identificat de cheia primară "ID\_Angajat" (NUMBER(7)), iar atributele precum "Nume\_Angajat" (VARCHAR2(100)), "Pozitie" (VARCHAR2(50)), "Data\_Angajare" (DATE), și "Salariu" (DECIMAL(10, 2)) rețin informații esențiale despre angajat și statutul său în cadrul bibliotecii.

EVENIMENTE

Tabelul "Evenimente" conține informații despre evenimentele organizate în bibliotecă. Cheia primară, "ID\_Eveniment" (NUMBER(7)), identifică în mod unic fiecare eveniment, iar atributele "Nume\_Eveniment" (VARCHAR2(100)), "Data\_Eveniment" (DATE), "Locatie" (VARCHAR2(100)), și "ID\_AngajatResponsabil" (NUMBER(7)) stochează detalii despre numele evenimentului, data, locație și angajatul responsabil. Cheia externă "ID\_AngajatResponsabil" este conectată la cheia primară a tabelului "Angajati\_B".

3.Diagrama Conceptuala

* 1. Reprezentare



3.2 Descrierea constrângerilor de integritate

CARTE

-ID\_CARTE - Primary Key

-ID\_EDITURA – Foreign Key – face legatura cu tabelul Editura

-ID\_SECTIUNE – Foreign Key – face legatura cu tabelul Sectiune

-ID\_EDITURA – NOT NULL – o carte trebuie sa aiba o editura

-ID\_SECTIUNE – NOT NULL – o carte trebuie sa se afle intr-o sectiune

ABONAMENT

-ID\_ABONAMENT – PRIMARY KEY

CITITOR

-ID\_CITITOR – PRIMARY KEY

-ID\_ABONAMENT – FOREIGN KEY – face legatura cu tabelul Abonament

-ID\_ABONAMENT – NOT NULL – un cititor trebuie sa aibe obligatoriu un abonament

IMPRUMUTURI

-ID\_IMPRUMUTURI – PRIMARY KEY

-ID\_CITITOR – FOREIGN KEY – face legatura cu tabelul Cititor

-ID\_CARTE – FOREIGN KEY – face legatura cu tabelul Carte

-ID\_CITITOR – NOT NULL – Un imprumut trebuie sa fie facut obligatoriu de un cititor

-ID\_CARTE – NOT NULL – Un imprumut trebuie sa fie asociat obligatoriu unei carti

EDITURA

-ID\_EDITURA – PRIMARY KEY

SECTIUNE

-ID\_SECTIUNE – PRIMARY KEY

-ID\_ANGAJATRESPONSABIL – FOREIGN KEY – face legatura cu tabelul Angajat

-ID\_ANGAJATRESPONSABIL – NOT NULL – o sectiune trebuie sa aiba obligatoriu un angajat care se ocupa de ea

ANGAJAT

-ID\_ANGAJAT – PRIMARY KEY

EVENIMENTE

-ID\_EVENIMENT – PRIMARY KEY

-ID\_ANGAJATRESPONSABIL – FOREIGN KEY – face legatura cu tabelul Angajat

-ID\_ANGAJATRESPONSABIL – NOT NULL – un eveniment trebuie sa fie organizat obligatoriu de un angajat

3.3 Scheme rationale

CARTE(id\_carte# , id\_editura FK, id\_sectiune FK, titlu , autor)

ABONAMENT(id\_abonament# , tip\_abonament, cost\_abonament)

CITITOR(id\_cititor# , id\_abonament FK, nume\_cititor, adresa, email)

EDITURA(id\_editura# , nume\_editura, tara\_sediu)

ANGAJATI(id\_angajat# , nume\_angajat, pozitie, data\_angajare, salariu)

EVENIMENTE(id\_evenimente# , id\_angajatresponsabil FK, nume\_eveniment, data\_eveniment, locatie)

SECTIUNE (id\_sectiune# , id\_angajatresponsabil FK, nume\_sectiune)

IMPRUMUTURI(id\_imprumut# , id\_cititor FK, id\_carte FK, data\_imprumut , data\_restituire)

4. Implementarea modelului într-un sistem de gestiune a bazelor de date

4.1 Crearea tabelelor si introducerea datelor

CREATE TABLE Abonamente (

ID\_Abonament NUMBER(7) PRIMARY KEY,

Tip\_Abonament VARCHAR(50),

Cost\_Abonament DECIMAL(10, 2)

);

INSERT INTO Abonamente (ID\_Abonament, Tip\_Abonament, Cost\_Abonament) VALUES

(1, 'Abonament Lunar', 15.99);

INSERT INTO Abonamente (ID\_Abonament, Tip\_Abonament, Cost\_Abonament) VALUES

(2, 'Abonament Anual', 150.00);

Select \* From Abonamente;

CREATE TABLE Cititori (

ID\_Cititor NUMBER(7) PRIMARY KEY,

Nume\_Cititor VARCHAR(100),

Adresa VARCHAR(255),

Email VARCHAR(100),

ID\_Abonament INT NOT NULL,

CONSTRAINT fk\_abonament FOREIGN KEY (ID\_Abonament) REFERENCES Abonamente(ID\_Abonament)

);

INSERT INTO Cititori (ID\_Cititor, Nume\_Cititor, Adresa, Email, ID\_Abonament) VALUES

(1, 'Adrian Ionut', 'Bulevardul Camile Ressu, Nr. 42', 'adrian.ionut@gmail.com', 1);

INSERT INTO Cititori (ID\_Cititor, Nume\_Cititor, Adresa, Email, ID\_Abonament) VALUES

(2, 'Cristescu Andrei', 'Bulevardul Theodor Pallady, Nr. 7', 'cristescu.andrei@yahoo.com.com', 2);

INSERT INTO Cititori (ID\_Cititor, Nume\_Cititor, Adresa, Email, ID\_Abonament) VALUES

(3, 'Munteanu Gabriel', 'Bulevardul Nicolae Grigorescu, Nr. 42', 'munteanu.gabriel@gmail.com', 1);

INSERT INTO Cititori (ID\_Cititor, Nume\_Cititor, Adresa, Email, ID\_Abonament) VALUES

(4, 'Budau Fabian', 'Bulevardul Theodor Pallady, Nr. 66', 'budau.fabian@yahoo.com.com', 1);

Select \* From Cititori;

CREATE TABLE Editura (

ID\_Editura NUMBER(7) PRIMARY KEY,

Nume\_Editura VARCHAR(100),

Tara\_Sediu VARCHAR(50)

);

INSERT INTO Editura (ID\_Editura, Nume\_Editura, Tara\_Sediu) VALUES

(1, 'Editura Bibliophile', 'Romania');

INSERT INTO Editura (ID\_Editura, Nume\_Editura, Tara\_Sediu) VALUES

(2, 'Editura Canonica', 'Romania');

Select \* From Editura;

CREATE TABLE Angajati\_B (

ID\_Angajat NUMBER(7) PRIMARY KEY,

Nume\_Angajat VARCHAR(100),

Pozitie VARCHAR(50),

Data\_Angajare DATE,

Salariu DECIMAL(10, 2)

);

INSERT INTO Angajati\_B (ID\_Angajat, Nume\_Angajat, Pozitie, Data\_Angajare, Salariu) VALUES

(1, 'Alex Dinescu', 'Bibliotecar', TO\_DATE('2020-01-01', 'YYYY-MM-DD'), 2500.00);

INSERT INTO Angajati\_B (ID\_Angajat, Nume\_Angajat, Pozitie, Data\_Angajare, Salariu) VALUES

(2, 'Andra Demircan', 'Manager Sectiune', TO\_DATE('2019-03-15', 'YYYY-MM-DD'), 3000.00);

INSERT INTO Angajati\_B (ID\_Angajat, Nume\_Angajat, Pozitie, Data\_Angajare, Salariu) VALUES

(3, 'Filip Popa', 'Ingrijitor', TO\_DATE('2012-09-19', 'YYYY-MM-DD'), 1900.00);

INSERT INTO Angajati\_B (ID\_Angajat, Nume\_Angajat, Pozitie, Data\_Angajare, Salariu) VALUES

(4, 'Denis Botea', 'Manager Sectiune', TO\_DATE('2021-11-09', 'YYYY-MM-DD'), 3000.00);

Select \* From Angajati\_B;

CREATE TABLE Evenimente (

ID\_Eveniment NUMBER(7) PRIMARY KEY,

Nume\_Eveniment VARCHAR(100),

Data\_Eveniment DATE,

Locatie VARCHAR(100),

ID\_AngajatResponsabil INT NOT NULL,

CONSTRAINT fk\_angajat\_eveniment FOREIGN KEY (ID\_AngajatResponsabil) REFERENCES Angajati\_B(ID\_Angajat)

);

INSERT INTO Evenimente (ID\_Eveniment, Nume\_Eveniment, Data\_Eveniment, Locatie, id\_angajatresponsabil) VALUES

(1, 'Lansare carte noua', TO\_DATE('2024-01-10', 'YYYY-MM-DD'), 'Sala de Lectura', 2);

INSERT INTO Evenimente (ID\_Eveniment, Nume\_Eveniment, Data\_Eveniment, Locatie, id\_angajatresponsabil) VALUES

(2, 'Sesiune de autografe', TO\_DATE('2024-01-17', 'YYYY-MM-DD'), 'Holul Bibliotecii', 2);

Select \* From Evenimente;

CREATE TABLE Sectiune (

ID\_Sectiune NUMBER(7) PRIMARY KEY,

Nume\_Sectiune VARCHAR(100),

ID\_AngajatResponsabil INT NOT NULL,

CONSTRAINT fk\_angajat\_sectiune FOREIGN KEY (ID\_AngajatResponsabil) REFERENCES Angajati\_B(ID\_Angajat)

);

INSERT INTO Sectiune (ID\_Sectiune, Nume\_Sectiune, ID\_AngajatResponsabil) VALUES

(1, 'Fictiune', 1);

INSERT INTO Sectiune (ID\_Sectiune, Nume\_Sectiune, ID\_AngajatResponsabil) VALUES

(2, 'Stiinte', 2);

INSERT INTO Sectiune (ID\_Sectiune, Nume\_Sectiune, ID\_AngajatResponsabil) VALUES

(3, 'Romance', 4);

SELECT \* FROM Sectiune;

CREATE TABLE Carte (

ID\_Carte NUMBER(7) PRIMARY KEY,

Titlu VARCHAR(255),

Autor VARCHAR(50),

ID\_Editura INT NOT NULL,

ID\_Sectiune INT NOT NULL,

CONSTRAINT fk\_editura FOREIGN KEY (ID\_Editura) REFERENCES Editura(ID\_Editura),

CONSTRAINT fk\_sectiune FOREIGN KEY (ID\_Sectiune) REFERENCES Sectiune(ID\_Sectiune)

);

INSERT INTO Carte (ID\_Carte, Titlu, Autor, ID\_Editura, ID\_Sectiune) VALUES

(1, 'Cronicile Bibliotecii', 'Gabriel Ionita' , 1, 1);

INSERT INTO Carte (ID\_Carte, Titlu, Autor, ID\_Editura, ID\_Sectiune) VALUES

(2, 'Colt Alb', 'Jack London', 2, 2);

INSERT INTO Carte (ID\_Carte, Titlu, Autor, ID\_Editura, ID\_Sectiune) VALUES

(3, 'Harry Potter Si Piatra Filozofala', 'J.K. Rowling', 1, 1);

INSERT INTO Carte (ID\_Carte, Titlu, Autor, ID\_Editura, ID\_Sectiune) VALUES

(4, '1984', 'George Orwell', 2, 1);

INSERT INTO Carte (ID\_Carte, Titlu, Autor, ID\_Editura, ID\_Sectiune) VALUES

(5, 'Mândrie si Prejudecata', 'Jane Austen', 1, 2);

INSERT INTO Carte (ID\_Carte, Titlu, Autor, ID\_Editura, ID\_Sectiune) VALUES

(6, 'Crima din Orient Express', 'Agatha Christie', 2, 2);

INSERT INTO Carte (ID\_Carte, Titlu, Autor, ID\_Editura, ID\_Sectiune) VALUES

(7, 'To Kill a Mockingbird', 'Harper Lee', 1, 3);

INSERT INTO Carte (ID\_Carte, Titlu, Autor, ID\_Editura, ID\_Sectiune) VALUES

(8, 'Cercul de piatră', 'Diana Gabaldon', 2, 3);

Select \* From Carte;

CREATE TABLE Imprumuturi(

ID\_Imprumut NUMBER(7) PRIMARY KEY,

ID\_Cititor INT NOT NULL,

ID\_Carte INT NOT NULL,

Data\_Imprumut DATE,

Data\_Restituire DATE,

CONSTRAINT fk\_cititor FOREIGN KEY (ID\_Cititor) REFERENCES Cititori(ID\_Cititor),

CONSTRAINT fk\_carte FOREIGN KEY (ID\_Carte) REFERENCES Carte(ID\_Carte)

);

INSERT INTO Imprumuturi (ID\_Imprumut, ID\_Cititor, ID\_Carte, Data\_Imprumut, Data\_Restituire) VALUES

(1, 1, 2, TO\_DATE('2024-01-5', 'YYYY-MM-DD'), TO\_DATE('2024-02-05', 'YYYY-MM-DD'));

INSERT INTO Imprumuturi (ID\_Imprumut, ID\_Cititor, ID\_Carte, Data\_Imprumut, Data\_Restituire) VALUES

(2, 2, 1, TO\_DATE('2024-01-09', 'YYYY-MM-DD'), TO\_DATE('2024-02-09', 'YYYY-MM-DD'));

INSERT INTO Imprumuturi (ID\_Imprumut, ID\_Cititor, ID\_Carte, Data\_Imprumut, Data\_Restituire) VALUES

(3, 3, 6, TO\_DATE('2024-01-09', 'YYYY-MM-DD'), TO\_DATE('2024-02-09', 'YYYY-MM-DD'));

INSERT INTO Imprumuturi (ID\_Imprumut, ID\_Cititor, ID\_Carte, Data\_Imprumut, Data\_Restituire) VALUES

(4, 2, 4, TO\_DATE('2024-01-10', 'YYYY-MM-DD'), TO\_DATE('2024-02-10', 'YYYY-MM-DD'));

Select \* From Imprumuturi;

4.2 Ștergerea tabelelor si constrângerilor

DROP TABLE CARTE CASCADE CONSTRAINTS;

DROP TABLE ABONAMENTE CASCADE CONSTRAINTS;

DROP TABLE CITITORI CASCADE CONSTRAINTS;

DROP TABLE IMPRUMUTURI CASCADE CONSTRAINTS;

DROP TABLE EDITURA CASCADE CONSTRAINTS;

DROP TABLE SECTIUNE CASCADE CONSTRAINTS;

DROP TABLE ANGAJATI\_B CASCADE CONSTRAINTS;

DROP TABLE EVENIMENTE CASCADE CONSTRAINTS;